NPS69-86-008

NAVAL POSTGRADUATE SCHOOL

Monterey, California

IMPROVEMENTS TO THE STANIFORTH-MITCHELL

BAROTROPIC NUMERICAL WEATHER PREDICTION CODE

R. E. NEWTON

October 1986

Interim Report for Period
June 1986 - September 1986

Approved for Public Release; Distribution Unlimited

Prepared for:

Naval Environmental Prediction Research Facility
Monterey, California 93943

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral R. C. Austin
Superintendent

D. A. Schrady
Provost

The work reported herein was supported by the Naval Environmental Prediction Research Facility.

This report was prepared by:

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S) NPS69-86-008	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) 69	7a. NAME OF MONITORING ORGANIZATION Naval Environmental Prediction Research Facility
6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943	7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943
8a. NAME OF FUNDING / SPONSORING ORGANIZATION (As in 7a) 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
IMPROVEMENTS TO THE STANIFORTH-MITCHELL BAROTROPIC NUMERICAL WEATHER PREDICTION CODE

12 PERSONAL AUTHOR(S)
R. E. Newton

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Interim	FROM 6/1/86 TO 9/30/86	15 October 1986	

16 SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Finite element; numerical weather prediction; generalized eigenvalue problem; Helmholtz equation

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Improvements to the Staniforth-Mitchell barotropic numerical weather prediction code are proposed. Three separate contributions are described: (1) a direct solution of the Helmholtz equation is offered as a substitute for the iterative scheme currently employed; (2) a new algorithm is developed to solve the generalized eigenvalue problem for symmetric tridiagonal matrices; (3) the latter algorithm is extended to deal with a periodic boundary condition. It is estimated that a direct solution of the Helmholtz equation, for a 13 x 13 grid, will effect a computation time saving per time step of at least 10 percent. FORTRAN 77 listings of the subroutines needed to implement the proposed improvements are included.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified
22a NAME OF RESPONSIBLE INDIVIDUAL R. E. Newton	22b TELEPHONE (Include Area Code) (408) 646-3424 22c OFFICE SYMBOL 69

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

IMPROVEMENTS TO THE STANIFORTH-MITCHELL BAROTROPIC NUMERICAL WEATHER PREDICTION CODE

Introduction

The barotropic finite element code described by Staniforth and Mitchell in Ref. 1 has been the point of departure for extensive efforts at the Naval Postgraduate School to develop improved techniques for numerical weather prediction. Two earlier reports (Refs. 2 & 3) described applications of tensor product techniques to extensions of the code developed by Hinsman (Ref. 4). The present report is concerned with proposed alterations of the Staniforth-Mitchell code for the purpose of improving computational efficiency.

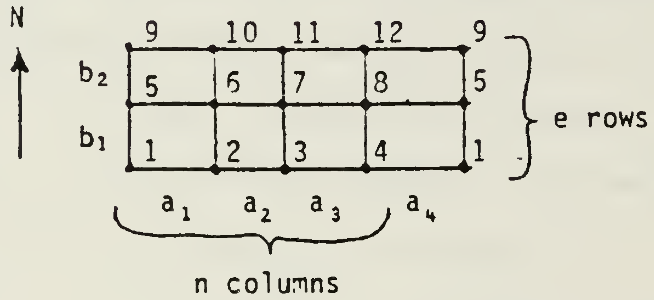
Three separate contributions are reported here. First of these is a scheme for the direct solution of the Helmholtz equation as a substitute for the Concus and Golub iterative algorithm (Ref. 5) used in the Staniforth-Mitchell code. The second item is an improved solution of the generalized eigenvalue problem that arises in transforming the two-dimensional Poisson and Helmholtz problems into a series of one-dimensional problems. This solution takes advantage of the special form of the matrices involved (symmetric, tridiagonal) and avoids matrix inversion. The third item deals with the special form of the generalized eigenvalue problem that arises when there is a periodic boundary condition in the east-west direction. In this instance the symmetric, formerly tridiagonal, matrices remain symmetric, but have nonzero elements in the upper right-hand and lower left-hand corners. A separate solution algorithm is required for determination of the eigenvalues and also for the eigenvectors.

Direct Solution of the Helmholtz Equation

The Helmholtz equation may be written in the form

$$\text{delsq } w - h\, w = f \qquad \langle 1 \rangle$$

where delsq is the two-dimensional Laplace operator, h is a function of y, f is a function of x and y, and w is the dependent variable. For the Finite Element (FE) discretization, consider the grid of Fig. 1. There are e east-west grid lines and n north-



(Note the periodic boundary condition in the E-W direction.)

Fig. 1. Node numbering and spacing.

south grid lines, defining ne intersections (nodes). Node numbering is from west to east, beginning at the southwest corner. The positive x direction is eastward and the positive y direction is northward. In the Galerkin FE discretization process the product h w is treated as a single entity and the tensor product concept is employed in writing the matrix form of the equation. The result is

$$MX\ W\ SY + SX\ W\ MY - MX\ W\ H\ MY = V \qquad \langle 2 \rangle$$

where MX and SX are n x n symmetric "mass" and "stiffness" matrices for the x direction, MY and SY are the corresponding e x e matrices for the y direction, W is an n x e matrix of nodal values of the dependent variable, H is diagonal e x e, and V is n x e. MX and SX depend only on the node spacing a_i and MY and SY depend only on the node spacing b_i. Defining equations for these matrices are given in Appendix A. Note that matrices SX and SY are the negatives of standard FE stiffness matrices. The

columns of W contain nodal values of w from the west-east rows, beginning with the most southerly. If the nodal values of f are similarly arranged in an n x e matrix F, then V is the product MX F MY. The nonzero entries in the diagonal matrix H are the values of h for the successive rows of nodes, beginning with the most southerly.

It is advantageous for the succeeding manipulations to transpose the terms of <2> to obtain

$$SY\ WT\ MX + MY\ WT\ SX - MY\ H\ WT\ MX = VT \qquad \langle 3 \rangle$$

where WT and VT are the respective transposes of W and V. Note that all of the other matrices are symmetric.

Consider the generalized eigenvalue problem

$$SX\ p_i = l_i\ MX\ p_i \qquad \langle 4 \rangle$$

where p_i is the ith eigenvector (n x 1) and l_i is the corresponding eigenvalue. Assembling the eigenvectors in an n x n modal matrix P and the eigenvalues in an n x n (diagonal) spectral matrix L, the result may be exhibited as

$$SX\ P = MX\ P\ L \qquad \langle 5 \rangle$$

It is advantageous to require that the eigenvectors be normalized so that PT MX P = I, where PT is the transpose of P and I is the nth order identity matrix. If <3> is postmultiplied by P and <5> is used to replace SX P in the second term, the result is

$$SY\ WT\ MX\ P + MY\ WT\ MX\ P\ L - MY\ H\ WT\ MX\ P = VT\ P \qquad \langle 6 \rangle$$

Let Q = WT MX P and substitute in <6> to get

$$SY'\ Q + MY\ Q\ L = U \qquad \langle 7 \rangle$$

where SY' = SY - MY H and U = VT P. Observe that SY' is not symmetric, but is tridiagonal. Now the ith column q_i of Q may be found by solving

$$(SY' + l_i MY) q_i = u_i \qquad \langle 8 \rangle$$

where u_i is the ith column of U. The solutions for the n q_i may be assembled into Q. It is easy to show that

$$W = P\ QT \qquad \langle 9 \rangle$$

where QT is the transpose of Q. This completes the solution.

Generalized Eigenvalue Problem for Symmetric Tridiagonal Matrices

Solution of the generalized eigenvalue problem <4> for the symmetric tridiagonal matrices SX and MX is based on the Sturm sequence property and the method of bisection. The theory is given by Wilkinson in Ref. 6. Using this theory and generalizing an ALGOL program constructed for the standard eigenvalue problem (Ref. 7), SUBROUTINE TRIEIG has been written to find the eigenvalues.

For any eigenvalue, the corresponding eigenvector can be found by rewriting <4> in the form

$$(SX - l_i MX) q_i = 0 \qquad \langle 10 \rangle$$

If the first component of q_i is chosen as unity, the first scalar equation of <10> determines the second component and each succeeding scalar equation determines an additional component. The vector thus found may then be normalized. SUBROUTINE EIGVEC performs these operations.

Once eigenvectors have been determined, SUBROUTINE RAYLEE may be used to refine the eigenvalues by evaluating the Rayleigh quotient for each eigenvector. Following this an additional call to SUBROUTINE EIGVEC enhances the accuracy of the eigenvectors. Experience to date indicates that further iterations are superfluous.

Generalized Eigenvalue Problem with Periodic Boundary Conditions

When there is a periodic boundary condition in the x direction, the tridiagonal form of matrices SX and MX is modified by the presence of nonzero elements in the upper right-hand and lower left-hand corners. The matrices remain symmetric, but the Sturm sequence and bisection procedure described above requires modification. Although the underlying strategy is unchanged, different software is required. In this instance, the eigenvalues are found by calling SUBROUTINE PEREIG. To determine the eigenvectors SUBROUTINE EIGVCP is called. A new problem arises here, because, if the grid spacing is uniform in the x direction, there will be double eigenvalues. Specifically, if n, the number of subdivisions in the x direction, is odd, there will be a single zero eigenvalue and all of the others will be double. If n is even, the eigenvalue of largest absolute value will also be single. Corresponding to double eigenvalues, the eigenvectors are not unique and special procedures must be used to construct vector pairs having the required orthogonality. SUBROUTINE EIGVCP tests for repeated eigenvalues and constructs the appropriate orthogonal pairs of vectors as required. For this periodic boundary condition, eigenvalue refinement via the Rayleigh quotient is effected by calling SUBROUTINE RAYLYP. An additional call to SUBROUTINE EIGVCP provides better eigenvectors. Further iteration is believed superfluous.

Remarks

Two additional subroutines are needed to utilize the proposed alterations to the Staniforth-Mitchell program. First of these is SUBROUTINE SETUP, a replacement for SUBROUTINE EBVSET. The second is SUBROUTINE SLVBVP, a replacement for SUBROUTINE EBVP2D. All of the new subroutines are listed in Appendix A.

The routines described herein have been tested and found to perform satisfactorily on a 12 x 12 grid. The parameter RF in TRIEIG and PEREIG (currently set at 1.0 E-07) may need to be adjusted for larger grids. In SUBROUTINE EIGVCP there is an additional parameter, SEP, which also may need adjustment for larger grids. It is currently set at 4.0 E-04.

There is a fundamental problem arising from the use of single precision arithmetic in solutions to the Helmholtz equation. This results from the fact that the mean geopotential height is approximately three orders of magnitude greater than the amplitude of a representative disturbance. Comparisons between single and double precision solutions show clearly that round-off adversely affects the former.

Operation counts have been conducted to evaluate the potential saving resulting from substituting the Helmholtz direct solver for the Concus-Golub iterative scheme. For a 13 x 13 grid and assuming only one cycle of iteration, there is a saving of 10 percent for each time step. For a 90 x 90 grid, the saving is reduced to 5 percent. If additional iterations are required for the Concus-Golub solution, these savings would, of course, be greater. The savings resulting from the proposed eigenvalue solutions have not been evaluated, but are expected to be substantial.

List of References

1. Staniforth, A. N., and H. L. Mitchell, "A Semi-Implicit Finite Element Barotropic Model," Monthly Weather Review, v. 105, p. 154-169, February 1977.

2. Newton, R. E., "Use of the Tensor Product for Numerical Weather Prediction by the Finite Element Method - Part 1," NPS69-84-001, Naval Postgraduate School, April 1984.

3. Newton, R. E., "Use of the Tensor Product for Numerical Weather Prediction by the Finite Element Method - Part 2," NPS69-84-005, Naval Postgraduate School, June 1984.

4. Hinsman, D. E., "Numerical Simulation of Atmospheric Flow on Variable Grids using the Galerkin Finite Element Method," Doctoral Dissertation, Naval Postgraduate School, March 1983.

5. Concus, P. and G. H. Golub, "Use of Fast Direct Methods for the Efficient Numerical Solution of Non-Separable Elliptic Equations," SIAM Journal of Numerical Analysis, v. 10, pp. 1103-1120.

6. Wilkinson, J. H., "The Algebraic Eigenvalue Problem," Oxford University Press, London, 1965.

7. Wilkinson, J. H., and C. Reinsch, "Linear Algebra," Springer, Berlin, 1971.

APPENDIX A. "MASS" and "STIFFNESS" MATRICES

$$
\underset{(n=4)}{MX} = \frac{1}{6}\begin{bmatrix} 2(a_4+a_1) & a_1 & 0 & a_4 \\ a_1 & 2(a_1+a_2) & a_2 & 0 \\ 0 & a_2 & 2(a_2+a_3) & a_3 \\ a_4 & 0 & a_3 & 2(a_3+a_4) \end{bmatrix}
$$

$$
\underset{(e=3)}{MY} = \frac{1}{6}\begin{bmatrix} 2b_1 & b_1 & 0 \\ b_1 & 2(b_1+b_2) & b_2 \\ 0 & b_2 & 2b_2 \end{bmatrix}
$$

$$
\underset{(n=4)}{SX} = \begin{bmatrix} -\frac{1}{a_4}-\frac{1}{a_1} & \frac{1}{a_1} & 0 & \frac{1}{a_4} \\ \frac{1}{a_1} & -\frac{1}{a_1}-\frac{1}{a_2} & \frac{1}{a_2} & 0 \\ 0 & \frac{1}{a_2} & -\frac{1}{a_2}-\frac{1}{a_3} & \frac{1}{a_3} \\ \frac{1}{a_4} & 0 & \frac{1}{a_3} & -\frac{1}{a_3}-\frac{1}{a_4} \end{bmatrix}.
$$

$$
\underset{(e=3)}{SY} = \begin{bmatrix} -\frac{1}{b_1} & \frac{1}{b_1} & 0 \\ \frac{1}{b_1} & -\frac{1}{b_1}-\frac{1}{b_2} & \frac{1}{b_2} \\ 0 & \frac{1}{b_2} & -\frac{1}{b_2} \end{bmatrix}
$$

Notes:

1. Matrices SX and SY are negatives of those called stiffness matrices in standard FE usage.

2. Forms given for MX and SX are for a periodic east-west boundary condition. For a Neumann boundary condition, the terms containing a_4 would be omitted.

APPENDIX B. FORTRAN LISTINGS

```
      SUBROUTINE SETUP(EIGVEC,EIGVAL,BIGE,BIGC,BIGA,S,HX,HY,HELM,DIR,
     1                 FOURTH,NI,NJ,CON)

************************************************************************
*  SUBROUTINE SETUP IS DESIGNED TO REPLACE EBVSET WHEN THE PERIODIC    *
*  BOUNDARY CONDITION IS IMPOSED ON EASTERN AND WESTERN BOUNDARIES     *
************************************************************************
C
C     AUTHOR: R. E. NEWTON, SUMMER 1986
C
C  ARGUMENTS
C     OUT - EIGVEC -  MATRIX OF EIGENVECTORS, NI X NI, IN X DIRECTION
C           EIGVAL -  VECTOR OF EIGENVALUES, NI, IN NONDESCENDING ORDER
C            BIGE  -  DIAGONAL FACTOR OF COEFFICIENT MATRIX FOR Y
C                     DIRECTION
C            BIGC  -  SUPERDIAGONAL OF UPPER UNIT TRIANGULAR FACTOR OF
C                     COEFFICIENT MATRIX FOR Y DIRECTION
C            BIGA  -  SUBDIAGONAL OF LOWER UNIT TRIANGULAR FACTOR OF
C                     COEFFICIENT MATRIX FOR Y DIRECTION
C     IN -     S   -  SQUARE OF MAP FACTOR, NI X NJ
C             HX   -  NODE SPACING IN X DIRECTION
C             HY   -  NODE SPACING IN Y DIRECTION
C            HELM  -  LOGICAL SWITCH - TRUE FOR HELMHOLTZ PROBLEM
C            DIR   -  LOGICAL SWITCH - TRUE FOR DIRICHLET B.C. ON
C                                      POISSON PROBLEM
C          FOURTH  -  LOGICAL SWITCH - TRUE FOR FOURTH ORDER SOLUTION OF
C                                      POISSON PROBLEM
C             NI   -  X-DIMENSION
C             NJ   -  Y-DIMENSION
C
C  NOTE:      N IS MAX(NI,NJ)
C-----------------------------------------------------------------------

      PARAMETER(N=13)
      REAL AS(N),BS(N),CS(N),S(NI,NJ),HX(NI),HY(NJ),
     1  AR(N),BR(N),CR(N),BIGA(NI,NJ),BIGC(NI,NJ),BIGE(NI,NJ),WU(N),
     2  EIGVEC(NI,NI),EIGVAL(NI),AM(N),BM(N),CM(N),CON
      LOGICAL HELM,DIR,FOURTH

      CF = 2.
      IF(FOURTH) CF = 5.

C  SET UP "MASS" MATRIX FOR X DIRECTION (SUBDIAGONAL AM, DIAGONAL BM,
C  SUPERDIAGONAL CM).   PERIODIC BOUNDARY CONDITION ASSUMED.   IF NOT
C  PERIODIC, CALL SETABC INSTEAD.

      CALL SETABX(AM,BM,CM,HX,CF,NI)

C  SET UP "STIFFNESS" MATRIX FOR X DIRECTION (SUBDIAGONAL AS, DIAGONAL
C  BS, SUPERDIAGONAL CS).  PERIODIC BOUNDARY CONDITION ASSUMED.  IF
C  NOT PERIODIC, CALL SETD2 INSTEAD.

      CALL SETD2N(AS,BS,CS,1.,HX,NI)

C  SOLVE THE GENERALIZED EIGENVALUE PROBLEM: K X = Z M X, WHERE K AND
```

```
C  M ARE STIFFNESS AND MASS MATRICES, RESPECTIVELY, AND X IS THE EIGEN-
C  VECTOR CORRESPONDING TO EIGENVALUE Z.  PERIODIC BOUNDARY CONDITIONS
C  ASSUMED.  IF NOT PERIODIC, CALL TRIEIG INSTEAD.

      CALL PEREIG(EIGVEC,EIGVAL,AS,BS,BM,AM,WU,NI)

C  SET UP "MASS" MATRIX (SUBDIAGONAL AM, DIAGONAL BM, SUPERDIAGONAL CM)
C  AND "STIFFNESS" MATRIX (SUBDIAGONAL AS, DIAGONAL BS, SUPERDIAGONAL
C  CS) FOR THE Y DIRECTION.

      CALL SETABC(AM,BM,CM,HY,CF,NJ)
      CALL SETD2(AS,BS,CS,1.,HY,NJ)
      IF(.NOT.HELM)GO TO 11

C  FOR THE HELMHOLTZ PROBLEM THE STIFFNESS MATRIX IS MODIFIED BY SUB-
C  TRACTING THE CORRESPONDING ELEMENT OF THE MASS MATRIX DIVIDED BY
C  (GZMN TIMES DELT**2 TIMES THE RELEVANT SQUARE OF THE MAP FACTOR).

      AS(1) = 0.
      BS(1) = BS(1) - BM(1)*CON/S(1,1)
      CS(1) = CS(1) - CM(1)*CON/S(1,2)
      NJM = NJ - 1
      DO 10 J=2,NJM
         AS(J) = AS(J) - AM(J)*CON/S(1,J-1)
         BS(J) = BS(J) - BM(J)*CON/S(1,J)
         CS(J) = CS(J) - CM(J)*CON/S(1,J+1)
10    CONTINUE
      AS(NJ) = AS(NJ) - AM(NJ)*CON/S(1,NJM)
      BS(NJ) = BS(NJ) - BM(NJ)*CON/S(1,NJ)
      CS(NJ) = 0.

C  LOOP OVER EIGENVALUES TO CONSTRUCT BIGA, BIGC, AND BIGE

11    DO 20 I = 1, NI
         EIG = EIGVAL(I)
         DO 12 J = 1, NJ
            AR(J) = AS(J) + EIG*AM(J)
            BR(J) = BS(J) + EIG*BM(J)
            CR(J) = CS(J) + EIG*CM(J)
12       CONTINUE
         IF(.NOT.HELM.AND..NOT.DIR.AND.I.EQ.NI)CR(1) = 0.

C  FACTOR COEFFICIENT MATRIX

         CALL SETTRI(BR,CR,AR,AR,BR,CR,NJ)

C  STORE RESULTS IN BIGE, BIGC AND BIGA.  NOTE SIGN CHANGES FOR BIGC
C  AND BIGA.

         DO 16 J = 1, NJ
            BIGE(I,J) = BR(J)
            BIGC(I,J) = -CR(J)
            BIGA(I,J) = -AR(J)
16       CONTINUE
20    CONTINUE
      RETURN
      END
```

```
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

      SUBROUTINE PEREIG(V,X,B,C,D,E,WU,N)

******************************************************************************
*  SUBROUTINE PEREIG USES THE METHOD OF BISECTION TO FIND EIGENVALUES        *
*  FOR THE GENERALIZED EIGENVALUE PROBLEM INVOLVING SYMMETRIC (ALMOST)       *
*  TRIDIAGONAL MATRICES WITH NONZERO ELEMENTS IN 'CORNERS' (PERIODIC         *
*  BOUNDARY CONDITION). IT IS ADAPTED FROM AN ALGOL PROGRAM NAMED            *
*  'BISECT' WRITTEN BY BARTH, MARTIN, AND WILKINSON (NUM. MATH. 9, 386-*
*  393 (1967)).  'BISECT' FINDS EIGENVALUES (STANDARD EIGENVALUE PROB-       *
*  LEM) FOR A SYMMETRIC TRIDIAGONAL MATRIX.  THE MATRICES FOR PEREIG         *
*  ARE INPUT AS VECTORS - C(N) AS THE DIAGONAL AND B(N) AS THE SUB-          *
*  DIAGONAL OF THE "STIFFNESS" MATRIX AND D(N) AS THE DIAGONAL AND E(N)*
*  AS THE SUBDIAGONAL OF THE "MASS" MATRIX.  B(1) AND E(1) ARE THE           *
*  "CORNER" ELEMENTS OF THE RESPECTIVE MATRICES.  SUBROUTINE EIGVCP          *
*  FINDS THE EIGENVECTORS AND NORMALIZES THEM WITH RESPECT TO THE MASS       *
*  MATRIX.  SUBROUTINE RAYLYP USES THR RAYLEIGH QUOTIENT TO OBTAIN IM-       *
*  PROVED EIGENVALUES USING THESE EIGENVECTORS.  A SECOND CALL TO            *
*  TO EIGVCP EFFECTS A CORRESPONDING IMPROVEMENT IN THE EIGENVECTORS.        *
******************************************************************************
C
C     AUTHOR: R. E. NEWTON, SUMMER 1986
C
C  ARGUMENTS
C     OUT -    V    -  MATRIX OF EIGENVECTORS, N X N, NORMALIZED WITH
C                      WITH RESPECT TO "MASS" MATRIX
C              X    -  VECTOR OF EIGENVALUES IN NONDESCENDING ORDER
C     IN  -    C    -  DIAGONAL OF "STIFFNESS" MATRIX
C              B    -  SUBDIAGONAL OF "STIFFNESS" MATRIX
C              D    -  DIAGONAL OF "MASS" MATRIX
C              E    -  SUBDIAGONAL OF "MASS" MATRIX
C              WU   -  WORK VECTOR
C              N    -  VECTOR SIZE (= NI)
C
C    NOTE  -   MATRICES ARE FOR X-DIRECTION WITH PERIODIC BOUNDARY
C              CONDITION.  NODE SPACING IS HX.
C
C-----------------------------------------------------------------------------
C
      INTEGER N,Z,I,A,K,N1
      REAL C(N),B(N),X(N),WU(N),EPS1,EPS2,RF,F1,XMIN,XMAX,
     1  X1,XU,XO,D(N),E(N),DC,DB,DD,DE,TMAX,TMIN,V(N,N),
     2  Q,R,S,Q1,R1,R2,DR,QTEMP,ROLD,SW
      DATA EPS1,RF/0.,1.E-7/
      N1=0
      Z=0

C  CALCULATION OF XMAX, XMIN

      DC=C(N)
      DB=ABS(B(N))
      DD=D(N)
      DE=E(N)
      XMAX=(DC+DB)/(DD+DE)
      XMIN=(DC-DB)/(DD-DE)
C  EIGENVALUES ASSUMED NEGATIVE.  IF EIGENVALUES ARE ALL POSITIVE,
```

```
C   REPLACE THE TWO PRECEDING LINES WITH THE FOLLOWING TWO LINES.
*       XMAX=(DC+DB)/(DD-DE)
*       XMIN=(DC-DB)/(DD+DE)
        DO 2 I=N-1,1,-1
           DC=C(I)
           DB=ABS(B(I))+ABS(B(I+1))
           DD=D(I)
           DE=E(I)+E(I+1)
           TMAX=(DC+DB)/(DD+DE)
           TMIN=(DC-DB)/(DD-DE)
C   EIGENVALUES ASSUMED NEGATIVE.  IF EIGENVALUES ARE ALL POSITIVE,
C   REPLACE THE PRECEDING TWO LINES WITH THE FOLLOWING TWO LINES.
*          TMAX=(DC+DB)/(DD-DE)
*          TMIN=(DC-DB)/(DD+DE)
           IF(TMAX.GT.XMAX)XMAX=TMAX
           IF(TMIN.LT.XMIN)XMIN=TMIN
2       CONTINUE

C   SET EPS2

        IF(XMIN+XMAX.GT.0.)THEN
            EPS2=RF*XMAX
        ELSE
            EPS2=RF*(-XMIN)
        END IF
        IF(EPS1.LE.0.)EPS1=EPS2
        EPS2=0.5*EPS1+7.*EPS2

C   INNER BLOCK

        XO=XMAX
        DO 4 I=1,N
            X(I)=XMAX
            WU(I)=XMIN
4       CONTINUE

C   LOOP FOR K-TH EIGENVALUE

        DO 100 K=N,1,-1
           XU=XMIN
           DO 6 I=K,1,-1
                IF(XU.LT.WU(I))THEN
                     XU=WU(I)
                     GO TO 10
                END IF
6          CONTINUE
10         IF(XO.GT.X(K))XO=X(K)
20         X1=(XU+XO)/2.DO
           Z=Z+1

C   STURM'S SEQUENCE

           A=0
           Q=C(1)-X1*D(1)
           Q1=C(2)-X1*D(2)
           R=B(1)-X1*E(1)
           S=C(N)-X1*D(N)
```

```
      R1=B(2)-X1*E(2)
      R2=0.
      DR=0.
      DO 30 I=2,N-2
           IF(Q.EQ.0.)THEN
                SW=1.
                IF(Q1+2.*R1.EQ.0.)SW=-1.
                Q=Q1+SW*2.*R1
                R1=R1+SW*Q1
                R2=SW*(B(I+1)-X1*E(I+1))
           END IF
           IF(Q.LT.0.)A=A+1
           S=S-R*R/Q
           ROLD=R
           R=DR-R1*R/Q
           QTEMP=Q1-R1*R1/Q
           R1=B(I+1)-X1*E(I+1)-R1*R2/Q
           Q1=C(I+1)-X1*D(I+1)-R2*R2/Q
           DR=-ROLD*R2/Q
           R2=0.D0
           Q=QTEMP
30    CONTINUE
      IF(Q.EQ.0.D0) THEN
           SW=1.D0
           IF(Q1+2.D0*R1.EQ.0.D0)SW=-1.D0
           Q=Q1+2.D0*SW*R1
           R1=R1+SW*Q1
           R=R+SW*(B(N)-X1*E(N))
           IF(Q.LT.0.D0)A=A+1
      ELSE
           IF(Q.LT.0.)A=A+1
      END IF
      S=S-R*R/Q
      R=B(N)-X1*E(N)-R*R1/Q
      Q1=Q1-R1*R1/Q
      IF(Q1.EQ.0.AND.R.NE.0.) THEN
           A=A+1
      ELSE
           IF(Q1.LT.0.)A=A+1
           IF(S-R*R/Q1.LT.0.)A=A+1
      END IF
      IF(A.LT.K)THEN
           IF(A.LT.1)THEN
                WU(1)=X1
                XU=X1
           ELSE
                WU(A+1)=X1
                XU=X1
                IF(X(A).GT.X1)X(A)=X1
           END IF
      ELSE
           XO=X1
      END IF
      IF(XO-XU.GT.2.*RF*(ABS(XU)+ABS(XO))+EPS1)GO TO 20
      X(K)=(XO+XU)/2.
100 CONTINUE
    CALL EIGVCP(V,B,C,D,E,X,N)
```

```
      CALL RAYLYP(X,V,B,C,WU,N)
      CALL EIGVCP(V,B,C,D,E,X,N)
      RETURN
      END

CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

      SUBROUTINE EIGVCP(V,B,C,D,E,X,N)

**************************************************************************
*  SUBROUTINE EIGVCP FINDS EIGENVECTORS BY DIRECT SOLUTION OF THE GOV-
*  ERNING LINEAR ALGEBRAIC EQUATIONS.  FOR ANY PAIR OF EQUAL EIGEN-
*  VALUES, THE CORRESPONDING VECTORS ARE MADE ORTHOGONAL WITH RESPECT
*  TO THE "MASS" MATRIX.  ALL VECTORS ARE NORMALIZED WITH RESPECT TO
*  THE "MASS" MATRIX.
**************************************************************************
C
C  AUTHOR  -  R. E. NEWTON, SUMMER 1986
C
C  ARGUMENTS
C     OUT -   V    -  MODAL MATRIX, N X N.  (COLUMNS ARE EIGENVECTORS.)
C     IN  -   B    -  SUBDIAGONAL OF STIFFNESS MATRIX
C             C    -  DIAGONAL OF STIFFNESS MATRIX
C             D    -  DIAGONAL OF MASS MATRIX
C             E    -  SUBDIAGONAL OF MASS MATRIX
C             X    -  VECTOR OF EIGENVALUES
C             N    -  VECTOR SIZE (=NI)
C
C-------------------------------------------------------------------------

      INTEGER J,K,N,L,N1
      PARAMETER(N1=12)
      REAL V(N,N),P(N1),Q(N1),T(N1),X(N),B(N),C(N),D(N),E(N),
     1     H,VN,TN,DIAG,X1,X2,VTMV,TTMV,TTMT,SUM

      L=0
      DO 30 K=1,N
         IF(L.NE.0)THEN
              L=0
              GO TO 30
         END IF
         X1=X(K)
         V(1,K)=1.D0
         V(2,K)=0.D0
         T(1)=0.D0
         T(2)=1.D0
         P(1)=B(1)-X1*E(1)
         P(2)=B(2)-X1*E(2)
         VN=-(C(1)-X1*D(1))/P(1)
         TN=-P(2)/P(1)
         DO 10 J=2,N-1
              DIAG=C(J)-X1*D(J)
              P(J+1)=B(J+1)-X1*E(J+1)
              V(J+1,K)=-(P(J)*V(J-1,K)+DIAG*V(J,K))/P(J+1)
              T(J+1)=-(P(J)*T(J-1)+DIAG*T(J))/P(J+1)
10       CONTINUE
```

```
C   CHECK FOR A DOUBLE ROOT

          IF(K.EQ.N)GO TO 101
          X2=X(K+1)
          CRIT=ABS((X2-X1)/(X2+X1))
          IF(CRIT.LT.4.E-4)GO TO 22

C   CONSTRUCT EIGENVECTOR FOR SINGLE ROOT

101       H=-(V(N,K)-VN)/(T(N)-TN)
          DO 11 J=2,N
               V(J,K)=V(J,K)+H*T(J)
11        CONTINUE

C   NORMALIZE WITH RESPECT TO MASS MATRIX

          P(1)=D(1)*V(1,K)+E(2)*V(2,K)+E(1)*V(N,K)
          DO 12 J=2,N-1
               P(J)=E(J)*V(J-1,K)+D(J)*V(J,K)+E(J+1)*V(J+1,K)
12        CONTINUE
          P(N)=E(N)*V(N-1,K)+D(N)*V(N,K)+E(1)*V(1,K)
          VTMV=0.DO
          DO 14 J=1,N
               VTMV=VTMV+P(J)*V(J,K)
14        CONTINUE
          VTMV=1.DO/SQRT(VTMV)
          DO 16 J=1,N
               V(J,K)=V(J,K)*VTMV
16        CONTINUE
          GO TO 30

C   CONSTRUCT EIGENVECTORS FOR DOUBLE ROOT AND NORMALIZE

22        L=1
          P(1)=D(1)*V(1,K)+E(2)*V(2,K)+E(1)*V(N,K)
          Q(1)=D(1)*T(1)+E(2)*T(2)+E(1)*T(N)
          DO 23 J=2,N-1
               P(J)=E(J)*V(J-1,K)+D(J)*V(J,K)+E(J+1)*V(J+1,K)
               Q(J)=E(J)*T(J-1)+D(J)*T(J)+E(J+1)*T(J+1)
23        CONTINUE
          P(N)=E(N)*V(N-1,K)+D(N)*V(N,K)+E(1)*V(1,K)
          Q(N)=E(N)*T(N-1)+D(N)*T(N)+E(1)*T(1)
          VTMV=0.DO
          TTMV=0.DO
          TTMT=0.DO
          DO 24 J=1,N
               VTMV=VTMV+P(J)*V(J,K)
               TTMV=TTMV+P(J)*T(J)
               TTMT=TTMT+Q(J)*T(J)
24        CONTINUE
          H=-TTMV/VTMV
          SUM=TTMT+2.DO*H*TTMV+H*H*VTMV
          VTMV=1.DO/SQRT(VTMV)
          SUM=1.DO/SQRT(SUM)
          H=H*SUM
          DO 26 J=1,N
               V(J,K+1)=V(J,K)*H+T(J)*SUM
```

```
            V(J,K)=V(J,K)*VTMV
26       CONTINUE
30    CONTINUE
      RETURN
      END

CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

      SUBROUTINE RAYLYP(X,V,B,C,P,N)

************************************************************************
*  SUBROUTINE RAYLYP USES THE RAYLEIGH QUOTIENT TO FIND IMPROVED
*  EIGENVALUES FROM THE ALREADY NORMALIZED EIGENVECTORS
************************************************************************
C
C  AUTHOR   -    R. E. NEWTON, SUMMER 1986
C
C  ARGUMENTS
C     OUT -   X     -  VECTOR OF N EIGENVALUES IN NONDESCENDING ORDER
C     IN  -   V     -  MODAL MATRIX, N X N. (NORMALIZED WITH RESPECT TO
C                      MASS MATRIX.)
C             B     -  SUBDIAGONAL OF STIFFNESS MATRIX
C             C     -  DIAGONAL OF STIFFNESS MATRIX
C             P     -  WORK VECTOR
C             N     -  VECTOR SIZE (= NI)
C
C-----------------------------------------------------------------------

      INTEGER J,K,N
      REAL V(N,N),P(N),X(N),B(N),C(N),X1

      DO 20 K=1,N
         P(1)=C(1)*V(1,K)+B(2)*V(2,K)+B(1)*V(N,K)
         DO 12 J=2,N-1
            P(J)=B(J)*V(J-1,K)+C(J)*V(J,K)+B(J+1)*V(J+1,K)
12       CONTINUE
         P(N)=B(N)*V(N-1,K)+C(N)*V(N,K)+B(1)*V(1,K)
         X1=0.D0
         DO 16 J=1,N
            X1=X1+P(J)*V(J,K)
16       CONTINUE
         X(K)=X1
20    CONTINUE
      RETURN
      END

CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

      SUBROUTINE SLVBVP(PHI,RHS,EIGVEC,BIGE,BIGC,BIGA,WK,DIR,HELM,
     1                  NI,NJ)
***********************************************************************
*  SUBROUTINE IS A SUBSTITUTE FOR EBVP2D
***********************************************************************
C
C  AUTHOR     -    R. E. NEWTON, SUMMER 1986
C
C  ARGUMENTS
```

```
C     OUT -   PHI   -  SOLUTION
C     IN  -   RHS   -  RIGHT-HAND SIDE
C           EIGVEC  -  MATRIX OF EIGENVECTORS
C             BIGE  -  INVERSE OF DIAGONAL FACTOR OF COEFFICIENT MATRIX
C             BIGC  -  SUPERDIAGONAL OF UNIT UPPER TRIANGULAR FACTOR OF
C                      COEFFICIENT MATRIX
C             BIGA  -  SUBDIAGONAL OF UNIT LOWER TRIANGULAR FACTOR OF
C                      COEFFICIENT MATRIX
C              WK   -  WORK AREA
C              DIR  -  LOGICAL SWITCH - TRUE FOR DIRICHLET B.C.
C             HELM  -  LOGICAL SWITCH - TRUE FOR HELMHOLTZ EQUATION
C              NI   -  X-DIMENSION
C              NJ   -  Y-DIMENSION
C------------------------------------------------------------------------------

      REAL PHI(NI,NJ),RHS(NI,NJ),EIGVEC(NI,NI),BIGE(NI,NJ),BIGC(NI,NJ),
     1  BIGA(NI,NJ),WK(NJ,NI),DUM
      LOGICAL DIR, HELM

C  TRANSFORM RIGHT-HAND SIDE

      CALL MATPR(WK,RHS,EIGVEC,NJ,NI,NI,NJ,NI,NI,DUM,DUM,DUM,DIR,
     1           .FALSE.)

C  PERFORM FORWARD REDUCTIONS

      DO 5 I = 1, NI
         WK(1,I) = WK(1,I)*BIGE(I,1)
5     CONTINUE
      DO 15 I = 1, NI
         DO 10 J = 2, NJ
              WK(J,I) = WK(J,I)*BIGE(I,J) + WK(J-1,I)*BIGA(I,J)
10       CONTINUE
15    CONTINUE

C  PERFORM BACK SUBSTITUTIONS

      DO 25 I = 1, NI
         DO 20 J = NJ-1, 1, -1
              WK(J,I) = WK(J,I) + WK(J+1,I)*BIGC(I,J)
20       CONTINUE
25    CONTINUE

C  BACK TRANSFORM RESULTS

      DO 40 J = 1, NJ
         DO 35 I = 1, NI
              DUM = 0.
              DO 30 K = 1, NI
                   DUM = DUM + EIGVEC(I,K)*WK(J,K)
30            CONTINUE
              PHI(I,J) = DUM
35       CONTINUE
40    CONTINUE
      RETURN
      END
```

```
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

      SUBROUTINE TRIEIG(V,X,B,C,D,E,WU,N)

**************************************************************************
*  SUBROUTINE TRIEIG USES THE METHOD OF BISECTION TO FIND EIGENVALUES *
*  FOR THE GENERALIZED EIGENVALUE PROBLEM INVOLVING SYMMETRIC TRIDIA-
*  GONAL MATRICES.  THE ROUTINE IS ADAPTED FROM AN ALGOL PROGRAM NAMED*
*  'BISECT' WRITTEN BY BARTH, MARTIN, AND WILKINSON (NUM. MATH. 9, 386-
*  393 (1967)).  'BISECT' FINDS EIGENVALUES (STANDARD EIGENVALUE PROB-
*  LEM) FOR A SYMMETRIC TRIDIAGONAL MATRIX.  THE MATRICES FOR TRIEIG
*  ARE INPUT AS VECTORS - C(N) AS THE DIAGONAL AND B(N) AS THE SUB-   *
*  DIAGONAL OF THE "STIFFNESS" MATRIX AND D(N) AS THE DIAGONAL AND E(N)
*  AS THE SUBDIAGONAL ON THE "MASS" MATRIX.  SUBROUTINE EIGVEC FINDS
*  THE EIGENVECTORS AND NORMALIZES THEM WITH RESPECT TO THE MASS MATRIX
**************************************************************************
C
C     AUTHOR: R. E. NEWTON, SUMMER 1985
C
C  ARGUMENTS
C     OUT -   V    -  MATRIX OF EIGENVECTORS, N X N, NORMALIZED WITH
C                     RESPECT TO THE "MASS" MATRIX
C             X    -  VECTOR OF EIGENVALUES IN NONDESCENDING ORDER
C     IN -    C    -  DIAGONAL OF "STIFFNESS" MATRIX
C             B    -  SUBDIAGONAL OF "STIFFNESS" MATRIX
C             D    -  DIAGONAL OF "MASS" MATRIX
C             E    -  SUBDIAGONAL OF "MASS" MATRIX
C             WU   -  WORK VECTOR
C             N    -  VECTOR SIZE (= NI)
C
C-------------------------------------------------------------------------
C
      INTEGER N,M1,N,Z,I,A,K
      REAL C(N),B(N),X(N),WU(N),EPS1,EPS2,RF,F1,XMIN,XMAX,
     1Q,X1,XU,XO,DELT,D(N),E(N),DC,DB,DD,DE,TMAX,TMIN,V(N,N),G(N,N)
      DATA EPS1,RF/0.,1.E-7/

C  CALCULATION OF XMAX, XMIN

      B(1)=0.
      E(1)=0.
      DC=C(N)
      DB=ABS(B(N))
      DD=D(N)
      DE=E(N)
      XMAX=(DC+DB)/(DD+DE)
      XMIN=(DC-DB)/(DD-DE)
C  NEGATIVE EIGENVALUES ASSUMED.  IF EIGENVALUES ARE ALL POSITIVE,
C  REPLACE THE TWO PRECEDING LINES WITH THE FOLLOWING TWO LINES.
*     XMAX=(DC+DB)/(DD-DE)
*     XMIN=(DC-DB)/(DD+DE)
      DO 2 I=N-1,1,-1
         DC=C(I)
         DB=ABS(B(I))+ABS(B(I+1))
         DD=D(I)
         DE=E(I)+E(I+1)
         TMAX=(DC+DB)/(DD+DE)
```

```
         TMIN=(DC-DB)/(DD-DE)
C   NEGATIVE EIGENVALUES ASSUMED.  IF EIGENVALUES ARE ALL POSITIVE,
C   REPLACE THE TWO PRECEDING LINES WITH THE FOLLOWING TWO LINES.
*        TMAX=(DC+DB)/(DD-DE)
*        TMIN=(DC-DB)/(DD+DE)
         IF(TMAX.GT.XMAX)XMAX=TMAX
         IF(TMIN.LT.XMIN)XMIN=TMIN
2      CONTINUE

C   SET EPS2

       IF(XMIN+XMAX.GT.0.)THEN
           EPS2=RF*XMAX
       ELSE
           EPS2=RF*(-XMIN)
       END IF
       IF(EPS1.LE.0.)EPS1=EPS2
       EPS2=0.5*EPS1+7.*EPS2

C   INNER BLOCK

       XO=XMAX
       DO 4 I=N,1,-1
           X(I)=XMAX
           WU(I)=XMIN
4      CONTINUE

C   LOOP FOR K-TH EIGENVALUE

       DO 100 K=N,1,-1
          XU=XMIN
          DO 6 I=K,1,-1
               IF(XU.LT.WU(I))THEN
                    XU=WU(I)
                    GO TO 10
               END IF
   6      CONTINUE
10        IF(XO.GT.X(K))XO=X(K)
20        X1=(XU+XO)/2.
          Z=Z+1

C   STURM'S SEQUENCE

          A=0
          Q=1.
          DO 30 I=1,N
               IF(Q.EQ.0.)THEN
                    DELT=ABS(B(I)-X1*E(I))/RF
               ELSE
                    DELT=(B(I)-X1*E(I))**2/Q
               END IF
               Q=C(I)-X1*D(I)-DELT
               IF(Q.LT.0.)A=A+1
 30       CONTINUE
          IF(A.LT.K)THEN
               IF(A.LT.1)THEN
                    WU(1)=X1
```

```
                    XU=X1
               ELSE
                    WU(A+1)=X1
                    XU=X1
                    IF(X(A).GT.X1)X(A)=X1
               END IF
          ELSE
               XO=X1
         END IF
          IF(XO-XU.GT.2.*RF*(ABS(XU)+ABS(XO))+EPS1)GO TO 20
          X(K)=(XO+XU)/2.
100   CONTINUE
      CALL EIGVEC(V,B,C,D,E,X,N)
      CALL RAYLEE(V,B,C,X,N)
      CALL EIGVEC(V,B,C,D,E,X,N)
      RETURN
      END

CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

      SUBROUTINE EIGVEC(V,B,C,D,E,X,N)

***********************************************************************
*  SUBROUTINE EIGVEC FINDS EIGENVECTORS BY DIRECT SOLUTION OF THE GOV-
*  ERNING LINEAR ALGEBRAIC EQUATIONS AND NORMALIZES THEM WITH RESPECT
*  TO THE MASS MATRIX.
***********************************************************************
C
C  AUTHOR      -  R. E. NEWTON, SUMMER 1985
C
C  ARGUMENTS
C     OUT -    V     -  MODAL MATRIX, N X N.  (COLUMNS ARE EIGENVECTORS.)
C     IN  -    B     -  SUBDIAGONAL OF STIFFNESS MATRIX
C              C     -  DIAGONAL OF STIFFNESS MATRIX
C              D     -  DIAGONAL OF MASS MATRIX
C              E     -  SUBDIAGONAL OF MASS MATRIX
C              N     -  VECTOR SIZE (= NI)
C
C-----------------------------------------------------------------------
C
      INTEGER J,K,N
      REAL V(N,N),P(5),X(N),B(N),C(N),D(N),E(N),X1,DSQRT,SUM

      DO 20 K=1,N
         X1=X(K)
         V(1,K)=1.
         P(2)=B(2)-E(2)*X1
         V(2,K)=(D(1)*X1-C(1))/P(2)
         DO 10 J=2,N-1
              P(J+1)=B(J+1)-E(J+1)*X1
              V(J+1,K)=-(P(J)*V(J-1,K)+(C(J)-D(J)*X1)*V(J,K))/P(J+1)
10       CONTINUE

C  NORMALIZE WITH RESPECT TO MASS MATRIX

         P(1)=D(1)*V(1,K)+E(2)*V(2,K)
         DO 12 J=2,N-1
```

```
            P(J)=E(J)*V(J-1,K)+D(J)*V(J,K)+E(J+1)*V(J+1,K)
12       CONTINUE
         P(N)=E(N)*V(N-1,K)+D(N)*V(N,K)
         SUM=0.
         DO 14 J=1,N
            SUM=SUM+P(J)*V(J,K)
14       CONTINUE
         SUM=1./SQRT(SUM)
         DO 16 J=1,N
            V(J,K)=V(J,K)*SUM
16       CONTINUE
20    CONTINUE
      RETURN
      END

CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

      SUBROUTINE RAYLEE(V,B,C,X,N)

************************************************************************
*  SUBROUTINE RAYLEE USES THE RAYLIEGH QUOTIENT TO FIND IMPROVED EIGEN-*
*  VALUES FROM THE ALREADY NORMALIZED EIGENVECTORS.                    *
************************************************************************
C
C  AUTHOR      -  R. E. NEWTON, SUMMER 1985
C
C  ARGUMENTS
C     OUT -   X     -  VECTOR OF EIGENVALUES IN ASCENDING ORDER
C     IN  -   V     -  MODAL MATRIX, N X N.  (NORMALIZED WITH RESPECT TO
C                      MASS MATRIX)
C             B     -  SUBDIAGONAL OF STIFFNESS MATRIX
C             C     -  DIAGONAL OF STIFFNESS MATRIX
C             N     -  VECTOR SIZE (= NI)
C
C-----------------------------------------------------------------------

      INTEGER J,K,N
      REAL V(N,N),P(5),X(N),B(N),C(N),X1

      DO 20 K=1,N
         P(1)=C(1)*V(1,K)+B(2)*V(2,K)
         DO 12 J=2,N-1
            P(J)=B(J)*V(J-1,K)+C(J)*V(J,K)+B(J+1)*V(J+1,K)
12       CONTINUE
         P(N)=B(N)*V(N-1,K)+C(N)*V(N,K)
         X1=0.
         DO 16 J=1,N
            X1=X1+P(J)*V(J,K)
16       CONTINUE
         X(K)=X1
20    CONTINUE
      RETURN
      END
```

INITIAL DISTRIBUTION LIST

		Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Superintendent Naval Postgraduate School Monterey, California 93943 ATTN Code 0142 Library	2
3.	Commanding Officer Naval Environmental Prediction Research Facility Monterey, California 93943	10
4.	Professor R. T. Williams, Code 63Wu Naval Postgraduate School Monterey, California 93943	5
5.	Professor R. E. Newton, Code 69Ne Naval Postgraduate School Monterey, California 93943	10
6.	Professor A. L. Schoenstadt, Code 53Zh Naval Postgraduate School Monterey, California 93943	1
7.	Professor D. Salinas, Code 69Sa Naval Postgraduate School Monterey, California 93943	1
8.	Professor R. H. Franke, Code 53Fe Naval Postgraduate School Monterey, California 93943	1
9.	Doctor A. N. Staniforth Recherche en Prevision Numerique Atmospheric Environment Service Dorval, Quebec H9P 1J3 Canada	1
10.	Research Administration, Code 012A Naval Postgraduate School Monterey, California 93943	1
11.	Professor Beny Neta, Code 53Nd Naval Postgraduate School Monterey, California 93943	1